UNIVERSITÄT AUGSBURG

## An annotated data set for pose estimation of swimmers

Thomas Greif and Rainer Lienhart
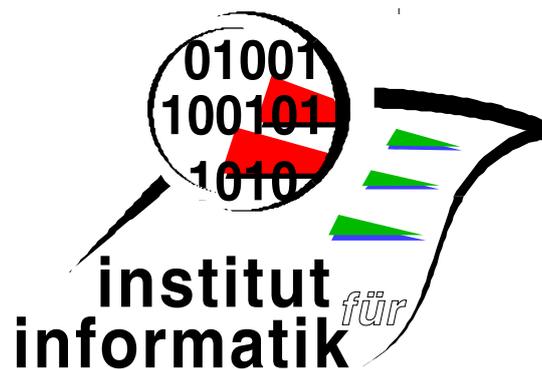
INSTITUT FÜR INFORMATIK

D-86135 AUGSBURG

AN ANNOTATED DATA SET FOR POSE ESTIMATION OF SWIMMERS

Anonymous ICME 2010 submission

Paper ID 1987

ABSTRACT

In this work we present an annotated data set for two–dimensional pose estimation of swimmers. The data set contains fifteen cycles of swimmers swimming backstroke with more than 1200 annotated video frames. A wide variety of subjects was used to create this data set, ranging from adult to teenage swimmers, both, male and female. For each frame of a cycle, the absolute positions of fourteen points corresponding to human joints were manually labeled.

The data set proves to be very challenging with respect to partial occlusions and high amounts of background noise, however, it does not contain any out–of–plane motions that would further complicate the task of full body pose estimation. It thus aims at pose estimation and pose tracking algorithms trying to advance the field of recovering human poses in videos with frequently missing parts and under difficult conditions.

We explain in detail the creation of the data set, discuss the difficulties we faced, and finally demonstrate how it is used to create a training data set containing normalized cycles for action–specific pose tracking.

Keywords— data set, pose estimation, pose tracking, human motion, swimmers

1. INTRODUCTION

Human pose estimation in images and videos is amongst the most challenging subjects of modern computer vision. Not only are human motions highly nonlinear, but they also appear in countless variations, ranging from different anatomical properties such as height or weight, over the clothes they wear, to the environment or background they appear in.

Pose estimation and pose tracking algorithms usually require a large set of training data to cover these aspects. Some techniques extract only visual information from it [1], others use a combination of visual and kinematic information to learn and detect different poses [2]. However, publicly available data sets, especially ones providing ground truth data for pose tracking in videos, are hard to find, so that most research groups build their own internal data sets to use with their own algorithms. While this aggravates the way for researchers new to the field, it also hinders the comparison of different algorithms and approaches. Thus,

benchmarking on a known data set is necessary to (1) obtain meaningful evaluations, (because only then it is possible to draw conclusions from the obtained results) and (2) to compare algorithms according to predefined criteria.

In this work we present a public data set for two–dimensional pose estimation of swimmers. The data set is intended for algorithms tracking the pose of swimmers, but also for general human pose estimation algorithms, trying to advance the field of accurate pose estimation and pose tracking under highly cluttered backgrounds and with frequently occluded body parts. We provide fifteen sequences of swimmers swimming backstroke with annotated ground truth data. The swimmers vary in age, sex and anatomical properties so that a wide range of possible motions and appearances is covered. Results can for example be compared by using a leave–one–out approach that estimates the average performance over all permutations by using the first fourteen sequences for training (1141 frames) and one sequence for testing (74 frames).

1.1. Related work

As mentioned above, there are only few publicly available data sets for human pose estimation. The most sophisticated full body data set is the HumanEva data set [3]. It features high–quality three–dimensional ground truth data captured with a commercial motion capture system, and contains sequences with out–of–plane motion and partial occlusion of several body parts. In addition to the ground truth and video data, it also provides code to evaluate the performance of pose tracking algorithms using their data set, which is done by a defined error measure. Although providing accurate motion data, it still was created under artificial laboratory settings and thus lacks variations in the appearance of the subjects typically observed in outdoor scenes. Ramanan's people data set [4] on the other hand covers a wide range of human appearances in images, but does not provide sequences that would contain crucial temporal information and thus cannot be used for pose tracking in videos. Ferrari's Buffy Stickmen data set [5] also provides humans in a variety of different locations and different appearances but lacks sequences as well as annotations for the lower half of the body, so this data set can only be used for upper body pose detection in still images.

2. RECORDING SETUP

2.1. Setup

It was our aim to create a natural and real world data set, showing swimmers in many varieties such as in public pools under bad lighting conditions and with a high amount of background activities. Contrary to controlled environments, public pools do not permit attaching a moving rack holding cameras and following the swimmers. We hence used a stationary camera setup with two cameras – one capturing everything above the water, and one capturing everything below. Several considerations led to the decision to use two cameras instead of one.

When recording with only a single camera, this camera has to be placed in a way that all parts of the swimmers are visible. Ideally, half of the lens thus should be under the water whereas the other half should be above the water. The main problem with this setup is that due to the movement of the water itself this cannot be achieved. Moreover splashes are likely to hit the upper part of the lens, introducing more noise than intended and possibly occluding parts of the swimmers, making an accurate annotation difficult.

By using two cameras we overcome the above problems because the cameras are mounted deep enough under and high enough above the water surface. However, new problems are introduced by such a setup. Firstly, both cameras have to be temporally synchronized. Fortunately, this is quite easy to achieve since we use two identical camera models and can therefore assume that the recording speeds do not differentiate noticeably. The problem of synchronization is thus reduced to finding the offset between both cameras, which can for example be determined by capturing a characteristic event and finding the first frame in both cameras where this event occurs. Secondly, the cameras have to be aligned horizontally as well as vertically. However, since the cameras remain stationary misalignments can be corrected once for all recordings easily. Lastly, the most significant drawback of a two–camera setup is that there will always be a spatial displacement and a displacement in scale between parts in the upper video and the ones in the lower video. This stems from the fact that the lens of the upper camera is placed above the water surface, and the optical axis does not coincide with the plane represented by the water surface. We therefore face the problem of loose joints, that is, we see certain joints twice, one time in the upper and and one time in the lower frame. For example, when the arm is pointing straight upwards, we see the shoulder in the upper video and the lower video. Section 3.2 explains how we handle frames with this property.

Due to the considerations above we chose a two–camera setup for capturing the swimmers. We used a self–designed rack that can hold both cameras and can easily be attached to the side of the pool. Figure 1 depicts a sketch of the used setup. Two typical frames this setup produces are depicted in figure 2.
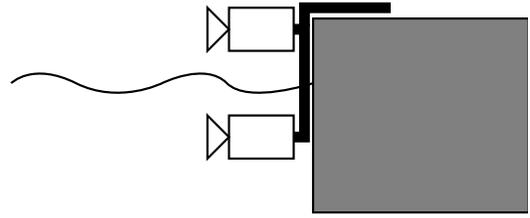
Fig. 1. Sketch of the used recording setup: two cameras were mounted to the rack that was placed in such a way that one camera only captured the parts under water and one camera only captured the parts above water.

2.2. Hardware

We used two identical Sony HDR-HC9E camcorders to capture the swimmers. While the lower camera was put into a waterproof casing and equipped with an "Ikelite W-20, 0.56x Wide-Angle Conversion Lens", the upper camera was equipped with a "Raynox HD-5050Pro Super Wideangle 0.5x" lens. With this combination of lenses we achieved the best horizontal coverage without a noticeable distortion.

3. DATA SET

The data set contains fifteen sequences/cycles of swimmers swimming backstroke. Each sequence corresponds to a different subject, captured at an open air pool with difficult lighting conditions using the setup described in section 2.1. In the following we will explain which points of each swimmer were labeled and how the annotations of the parts not facing the camera were created.

3.1. Annotations

We represent the human body by a set of fourteen unique points. Each of these points corresponds to a human joint: foot, knee, hip, center hip, neck, shoulder, elbow, and hand as depicted in figure 3. We manually labeled all of these points in each frame of the provided sequences to create the ground truth. The problem with recording humans from the side is, however, that most of the time the parts of the body not facing the camera are partially occluded. This means that in the majority of frames not all points are labeled, what however is significant for learning temporal relationships of the motion. We can overcome this by exploiting the fact that swimming is a cyclic motion, meaning that, depending on the swimming style, the motion of one half of the body tells us which motion to expect from the other half during a cycle. When swimming breast strokes, for example, one can assume that the second body half moves in the same way as the first, whereas at backstrokes, one can assume that the motion of the
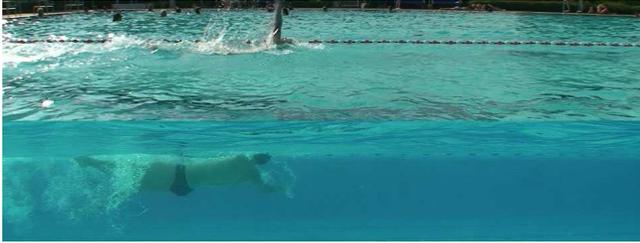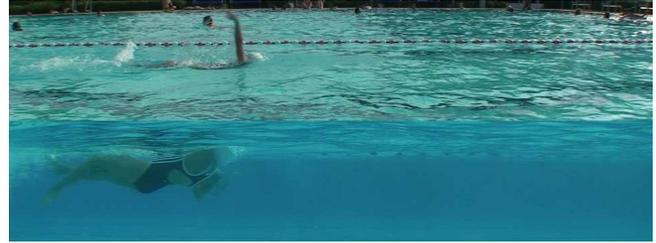
Fig. 2. The first frame of a sequence captured by the two–camera setup of (a) a male adult swimmer, and (b) a female teenage swimmer.

second body half is the motion of the first, delayed by exactly half of the cycle. This is subject to several assumptions, such as that the plane the spine lies in has to be perpendicular to the optical axis, however, it proves to be sufficiently correct for our data set. It is thus adequate to manually label the joints of only one half of the body. The other one can be reconstructed, given that we always annotate complete cycles.
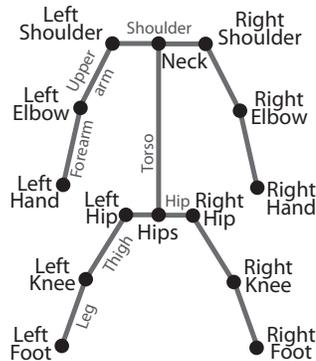

Fig. 3. The human body is represented by a simple stick figure model having eleven rigid parts (colored gray). The annotated points in each frame correspond to the human joints denoted by black dots.

3.2. Post processing

In section 2.1 we pointed out that by using a two–camera setup we face a displacement between joints appearing in the upper camera as well as in the lower camera. This did not influence the way we annotated points, because points occurring in the upper and lower frame were simply annotated in both frames. Such loose joints, however, result in a not fully connected stick figure model in certain frames. Put differently, the graph representing the annotations in the upper frame is not connected to the one representing the annotations in the lower frame. This presents a problem when intending to use only the motion data, that is, derive

kinematic relationships between the annotated points. In order to provide the motion data as well, we therefore need a fully connected stick figure model which is obtained by merging the annotations of the upper and lower frames. Figure 4 shows how this is done for a single frame. We start with the raw frames captured by both cameras as depicted in figure 4a. Clearly, we face the problem of loose joints in these frames, since the shoulder is visible in the upper and lower camera output. Merging the annotations is as simple as computing the translation vector between the points annotated twice and then translating all points in the upper frame that are connected to this point by this vector. However, before the translation vector can be computed, the annotations of both frames have to be expressed in coordinates of the same image coordinate system. We thus express all annotations by the image coordinate system of the combined frame which is created by simply combining the upper and the lower frame. The translation vector can then be computed as shown in figure 4b. Finally, figure 4c points out how the connected points are translated by the translation vector in order to obtain the fully connected stick figure model in this frame.

All in all 1215 frames were labeled and post processed in the manner described above. For each sequence we created three annotation lists: one containing only annotations from the upper camera, one containing only annotations from the lower camera, and a third annotation list with the merged annotations. Note that the specified point locations in this last (i.e. third) list do not necessarily concur with the locations of these points in the video frames. The annotations provided by this annotation list are therefore only intended to be used to derive motion parameters, not for extracting visual information. Furthermore, the output frames of each camera were merged into a single frame (in order to provide a single video file for each sequence) and all annotations are expressed in the coordinate system of this combined frame. Figure 5 shows an example of a frame and its annotations. Figure 5a depicts the annotations of the annotation list corresponding to the upper camera; only the parts visible above the water (in this case the
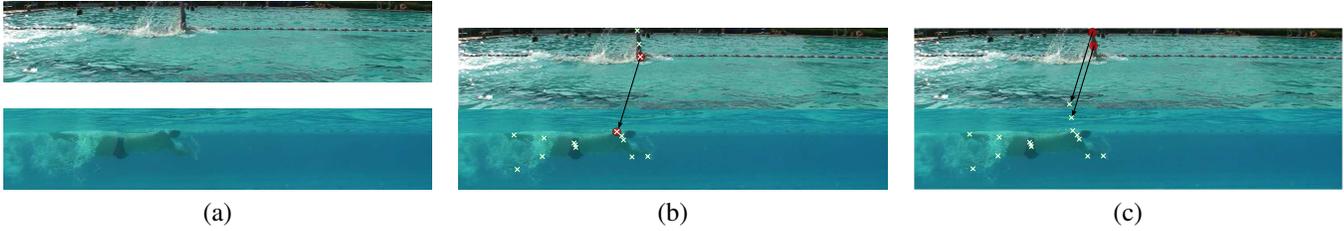
Fig. 4. Merging upper and lower annotations with loose joints: (a) the raw frames obtained from both cameras. (b) annotated points (white crosses) in both frames, now expressed in coordinates of the combined frame. The translation vector is computed between each pair of loose joints (red). (c) the merged annotations are obtained by translating all points connected to the loose joint by the translation vector.

upper arm) are annotated, whereas figure 5b shows only annotations of the parts under the water, thus corresponds to the annotation list of the lower camera. Finally, figure 5c shows the merged annotations.

3.3. Statistics

The whole data set consists of fifteen sequences where each sequence corresponds to exactly one cycle of backstroke swimming. Moreover, fifteen different subjects were used in order to create a generic data set. This is reflected by the length of each cycle and the size of each subject (in pixels). The mean length of all sequences is 81 frames with a variance of 98.7143 which corresponds to a standard deviation of 9.9355 frames.

The average size of a swimmer in a frame is 358.0667 pixels with a variance of 2021.6 and thus a standard deviation of 44.9626 pixels. By size we mean the distance (in pixels) between the left and right border of the bounding box surrounding each subject, maxed over all frames of the sequence the subject appears in.

We can see that the size of each swimmer as well as the length of each sequence differ significantly throughout all annotated cycles.

3.4. Data set structure

The video data of each sequence was recorded and stored as HDV MPEG–2 . In addition to the MPEG–2 source for each cycle, we provide the data as an AVI file compressed in motion JPEG. There exists one directory for each sequence, containing the video data as well as the annotation lists for the upper and lower camera, and the merged annotations (see section 3.2) for the motion data only. The annotation lists have XML format and include information about the video file itself (such as length, width and height) as well as the annotated points for each frame. Annotated points are represented in the file by their coordinates in the image and an id which is a simple zero–based identifier that algorithms can use to infer the points' identities. A snippet of such an annotation file is

shown in figure 6. The directory of a sequence denoted by # thus contains the following:

|	
s#.m2v	HDV MPEG–2 video source
s#.avi	AVI (M-JPEG codec) video source
s#_lower.al	lower camera annotations
s#_upper.al	upper camera annotations
s#_merged.al	merged annotations

4. APPLICATION

In this section we present an example of how the data set can be used for action–specific pose tracking in videos. We describe how the annotated frames are processed to create normalized cycles that serve as training data for a specific action - in this case backstroke swimming.

4.1. Training data for action–specific pose tracking

Action–specific pose tracking algorithms [6, 7, 8] normally need several samples of the action they later try to recognize. These algorithms infer their internal motion model parameters from data specifying the pose at specific times during one action. Speaking in terms of backstroke swimming, we assume that the action equals exactly one cycle of the backstroke motion, that is, an action starts with the arm pointing straight upwards and ends with the same pose. Since we are only interested in the motions that form an action, we only make use of the merged annotation lists not containing loose joints.

Rather than the absolute positions of the annotated points in a frame, we are interested in the angles between these points since they define the pose independent of the exact location of the swimmer. The first step in creating the training data is thus to convert the positions to angles ϕ with $0 \le \phi \le 2\pi$. Due to the angles' periodicity jumps can occur between consecutive values. In order to eliminate these, we add multiples of $\pm 2\pi$ whenever absolute jumps between
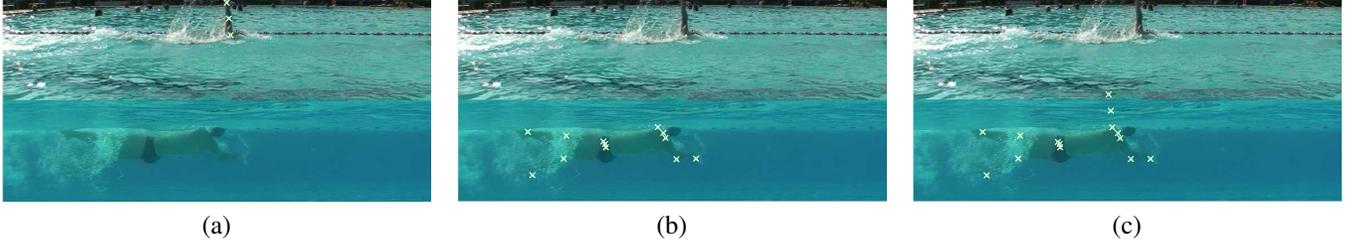
Fig. 5. Example of one completely annotated frame of the data set: (a) the annotation list corresponding to the upper camera only contains the subset of points visible in the upper frame. (b) the annotation list corresponding to the lower camera only contains the subset of points visible in the lower frame. (c) the merged annotation list contains all annotated points, however, not all locations concur with the video data, since points in the upper frame are translated to the underwater frame to obtain a fully connected stick figure model.

```
<?xml version="1.0" ?>
<annotationlist>
  <video>
    <file>./s1.avi</file>
    <length>93</length>
    <width>960</width>
    <height>360</height>
  </video>
  <annotation frame="1">
    <point>
      <id>5</id>
      <x>131</x>
      <y>314</y>
    </point>
    . . .
    <point>
      <id>13</id>
      <x>365</x>
      <y>239</y>
    </point>
  </annotation>
  <annotation frame="2">
    . . .
  </annotation>
</annotationlist>
```

Fig. 6. Snippet of an annotation file in XML format. Each annotation element corresponds to a single frame and contains several annotated points, where each point is defined by its location and a unique identifier.

the values of two frames are greater than or equal to π radians.

In order to train the model we first need to normalize the sequences to the same length, since any two recorded cycles of one action do almost never feature the same number of frames, and we want each position within the sequences represent the same progress within a cycle. Normalization is carried out by resampling by means of cubic spline interpolation and then sampling the required number of values. When interpolating, we face difficulties at the boundaries, that is, the beginning and the end of each cycle's function, due to missing values. To overcome this, we concatenate all fifteen cycles so that the outcome can be regarded as one single function for each angle. This resulting function then describes the motion of the angles for fifteen cycles. We let the beginning and the end of this function overlap for a certain number of frames in order to eliminate the problem of missing values. However, the first and the last pose of each sequence are alike but not the same (since each sequence corresponds to a different subject), which results in small discontinuities at the points where two cycles were concatenated. We smooth these by applying a simple low–pass filter to the concatenated sequences before resampling.

When we divide the concatenated cycles into sequences of the same normed length, we obtain the final training data that can be used to train an action–specific motion model. Figure 7 shows the temporal progress of two joints of five different sequences after resizing them to a length of 50 frames and smoothing. We see that although being from five different subjects the underlying motion is very similar and thus the cycles present suitable training data for action–specific models. The described training sequences are for example used in [9] to train a motion model that serves as temporal prior in a Bayesian tracking framework. They present a simple kinematic model that only relies on accelerations computed from the training data described in this section and show that such a simple model can exploit the full state dimension while still being computationally effective and achieving low errors. The training sequences can further be used for automatic pose initialization as demonstrated in [10]. They train a classifier of the upper arm using our provided videos and joint positions. This classifier is then applied to several test videos in order to find the location of the upper arm, and once found, they initialize the pose according to this location and kinematic relationships extracted from our ground truth data.
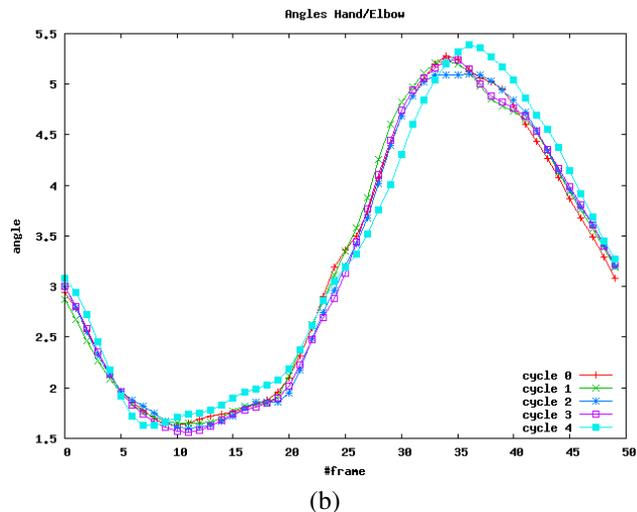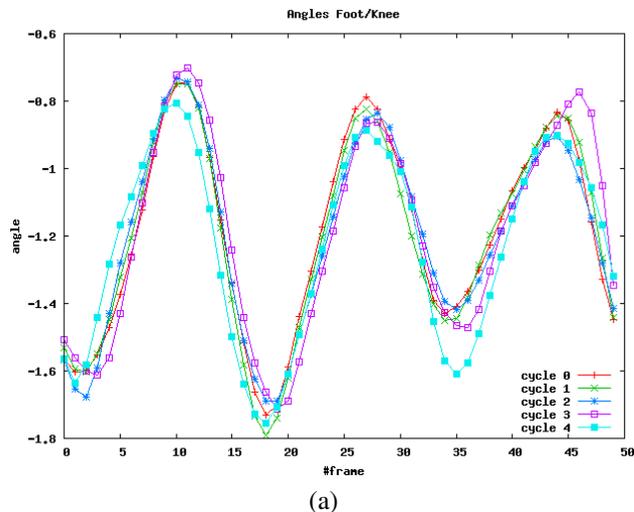
Fig. 7. Angles of (a) foot and (b) hand for five different cycles after normalizing the length to 50 frames and smoothing with a low–pass filter.

5. CONCLUSION AND FUTURE WORK

In this work, we have presented a new annotated data set for pose estimation of swimmers. It consists of more than 1200 manually labeled frames of swimmers swimming backstroke. Since the data set features many frames with partial occlusions as well as a very high amount of noise, it is intended for algorithms trying to advance the field of cluttered backgrounds and frequently missing parts. Joints of swimmers were annotated using a simple stick figure representation, and the second half of the body was reconstructed exploiting the fact that swimming is cyclic, so that occluded parts are also annotated. This is crucial for deriving kinematic relationships. We also demonstrated how the raw annotated cycles of the data set can be used to create training data for action–specific pose tracking.

In the future we will extend the data set in that we include annotations for all common swimming styles, and we will grow the data set by adding more subjects to the already annotated cycles.

6. REFERENCES

[1] Huazhong Ning, Wei Xu, Yihong Gong, and T. Huang, "Discriminative learning of visual words for 3d human pose estimation," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008, pp. 1–8.

[2] M. Andriluka, S. Roth, and B. Schiele, "Pictorial structures revisited: People detection and articulated pose estimation," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, June 2009.

[3] Leonid Sigal and Michael J. Black, "Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion," Tech. Rep., 2006.

[4] Deva Ramanan, "Learning to parse images of articulated bodies," in *In NIPS 2007*. 2006, NIPS.

[5] V. Ferrari, M. Marin-Jimenez, and A. Zisserman, "Progressive search space reduction for human pose estimation," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008.

[6] G. Rogez, J. Rihan, S. Ramalingam, C. Orrite, and P.H.S. Torr, "Randomized trees for human pose detection," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, June 2008, pp. 1–8.

[7] Ignasi Rius, Javier Varona, Jordi Gonzalez, and Juan J. Villanueva, "Action spaces for efficient bayesian tracking of human motion," in *ICPR '06: Proceedings of the 18th International Conference on Pattern Recognition*, Washington, DC, USA, 2006, pp. 472–475, IEEE Computer Society.

[8] D. Ormoneit, H. Sidenbladh, M. J. Black, and T. Hastie, "Learning and tracking cyclic human motion," in *In NIPS*. 2001, pp. 894–900, The MIT Press.

[9] T. Greif and R. Lienhart, "A kinematic model for bayesian tracking of cyclic human motion," in *IS&T/SPIE Electronic Imaging*, 2010.

[10] C. X. Ries and R. Lienhart, "Automatic pose initialization of swimmers in videos," in *IS&T/SPIE Electronic Imaging*, 2010.